

Catalog	Label	FullName	Description	Type	Default
10 ^x	10 ^x	10 to the power x	Raise 10 to the power in the X-register	Function (monadic)	
16-BIT ()	16-BIT	Word size 16 bits	Set word size to 16 bits for shortint	Setting (pgm)	OFF
1COMPL ()	1COMPL	1's complement	Set 1's complement mode for shortint (SBI depends on SBint)	Setting (pgm)	OFF
¹ / _x	¹ / _x	Reciprocal	Reciprocal (1/x)	Function (monadic)	
2COMPL (+)	2COMPL	2's complement	Set 2's complement mode for shortint (SBI depends on SBint)	Setting (pgm)	ON
2 ^x	2 ^x	2 to the power x	Raise 2 to the power in the X-register	Function (monadic)	
32-BIT ()	32-BIT	Word size 32 bits	Set word size to 32 bits for shortint	Setting (pgm)	OFF
3I×3Z	I×Z	Triple V = I × Z	X = R96 × R90 ; Y = R97 × R91 ; Z = R98 × R92	Command	
3V÷3I	V÷I	Triple Z = V / I	X = R93 / R96 ; Y = R94 / R97 ; Z = R95 / R98	Command	
3V÷3Z	V÷Z	Triple I = V / Z	X = R93 / R90 ; Y = R94 / R91 ; Z = R95 / R92	Command	
³ / _x	³ / _x	Cube root	Cube root of X	Function (monadic)	
64-BIT (+)	64-BIT	Word size 64 bits	Set word size to 64 bits for shortint	Setting (pgm)	ON
8-BIT ()	8-BIT	Word size 8 bits	Set word size to 8 bits for shortint	Setting (pgm)	OFF
ABS	ABS	Absolute	Magnitude (absolute value) of complex number	Function (legacy)	
ACOS	ARCCOS	Arc cosine	Inverse cosine	Function (monadic)	
ActUSB	ActUSB	Activate USB disk	Activate USB disk without exiting to DMCP (hardware only) (Does not autosave backup file to FAT)	Command	
AGM	AGM	Arithmetic Geometric Mean	Arithmetic geometric mean of X and Y	Function (dyadic)	
AGRAPH	AGRAPH	Alpha graphics	Show a 64 column pattern according to variable GRAMOD (OR: 0 ; SET: 1 ; OFF: 2 ; XOR: 3) (AGRAPH __ Tam menu)	Command	
ALL ₃ (+)	ALL	All digits notation	Set numeric display mode to ALL digits notation ; display all digits as far as possible with max. nn decimal zeros (ALL __ TamNonReg menu)	Setting (pgm)	3
ALLF	ALLF	Set all (models)	Pre-selects all models (opposite from ResetF)	Command	
AND	AND	AND	Logical AND (bitwise)	Function (dyadic)	
ARCCOS	ARCCOS	Arc cosine	Inverse cosine	Function (legacy)	
arcosh	arcosh	Inverse hyperbolic cosine	Inverse hyperbolic cosine	Function (monadic)	
ARCSIN	ARCSIN	Arc sine	Inverse sine	Function (legacy)	
ARCTAN	ARCTAN	Arc tangent	Inverse tangent	Function (legacy)	
arsinh	arsinh	Inverse hyperbolic sine	Inverse hyperbolic sine	Function (monadic)	
artanh	artanh	Inverse hyperbolic tangent	Inverse hyperbolic tangent	Function (monadic)	
ASIN	ARCSIN	Arc sine	Inverse sine	Function (monadic)	
ASR	ASR	Arithmetic shift right	Arithmetic shift right by n (ASR __ TamNonReg menu)	Function (monadic)	
ASSIGN	ASSIGN	Assign	1. Assign function or menu to key (in USER mode) using ASN (fShifted [1]) + <function>/<menu> + <key> on the keyboard 2. assign character to soft button using ASN (on menu α) + <character> + <soft button> on menu Myα 3. assign function or menu to soft button using ASN (fShifted [1]) + <function>/<menu> + <soft button> on menu MyMenu or user defined menu (only functions can be assigned in the case of a user defined menu and it must be active already) 4. create user defined menu using ASN (fShifted [1]) + USER (fShifted [2]) + <menu-name> (this menu will appear in the menu catalog and can be deleted using DELITM) (Select function from keyboard, catalog or enter name manually ; select character from keyboard or any of the alpha menus ; select menu from catalog or enter name manually ; unassign using ASN + ENTER)	Command	
ATAN	ARCTAN	Arc tangent	Inverse tangent	Function (monadic)	
ATAN2	ATAN2	Expanded inverse tangent	Arc tangent of Y / X	Function (dyadic)	
AtoSYM	→ 012	Convert to symmetrical components	Convert 3-phase a, b, c in Z, Y, X to symmetrical components a0, a1, a2 in Z, Y, X (→ abc)	Function (triadic)	
BACK	BACK	Back	Jumps n steps back in program (BACK ___ TamNonReg menu)	Command (PEM)	
BASE _{HP} [+]	BASE _{HP}	HP style base	Set for the classic (HP) convention that all stack registers are changed at once when base mode is changed (shortint values only)	Setting	ON
BATT?	BATT?	Battery	Battery voltage ±1 mV	Command	
BCD []	BCD	Binary coded decimal	Set BCD mode for shortint	Setting	OFF
BC?	BC?	Bit clear?	Test if bit n is clear (BC? __ TamNonReg menu)	Function (monadic)	
BEEP	BEEP	Beep	Play beep sound (4 tones)	Command	
BeginP ()	Begin	TVM begin payments	Payments at the beginning of each period (time value of money) (SBI depends on SBtvm)	Setting (pgm)	OFF
BestF	BestF	Best Fit	Select best curve fit model (BestF ___ TamNonReg menu)	Command	
BestF?	BestF?	Best Fit Setting	User setting for which models are allowed in the LR analysis ; set in menu MODEL	Command	
BIN	BIN	Binary	Convert X to binary and toggle binary mode	Setting (pgm)	OFF
Binom _p	Binom _p	Binomial pdf	Binomial probability density function	Function (tbd)	
Binom _x	Binom _x	Binomial cdf (lower)	Binomial cumulative distribution (lower tail)	Function (tbd)	

Catalog	Label	FullName	Description	Type	Default
Binom _▲	Binom _▲	Binomial cdf (upper)	Binomial cumulative distribution (upper tail)	Function (tbd)	
Binom ⁻¹	Binom ⁻¹	Binomial (inverse)	Binomial probability inverse function	Function (tbd)	
B _n	B _n	B _n	Bernoulli number (new definition)	Function (monadic)	
B _n *	B _n *	B _n *	Bernoulli number (old definition)	Function (monadic)	
BS?	BS?	Bit set?	Test if bit n is set (BS? __ TamNonReg menu)	Function (monadic)	
BUZZ	BUZZ	Buzzer	Play sound (input from stack : frequency in Hz (0 = silent) ; duration in ms (max 2000))	Command	
B.SWP	B.SWP	Byte swap	Swap bytes	Function (monadic)	
CASE	CASE	Case	Case statement, skips number of steps (CASE normally followed by number of GTO's) (CASE __ Tam menu)	Command (PEM)	
CauchF []	CauchF	Cauchy fit	Cauchy curve fitting	Setting	OFF
Cauch _p	Cauch _p	Cauchy pdf	Cauchy probability density function	Function (tbd)	
Cauch _▲	Cauch _▲	Cauchy cdf (lower)	Cauchy cumulative distribution (lower tail)	Function (tbd)	
Cauch _▲	Cauch _▲	Cauchy cdf (upper)	Cauchy cumulative distribution (upper tail)	Function (tbd)	
Cauch ⁻¹	Cauch ⁻¹	Cauchy (inverse)	Cauchy probability inverse function	Function (tbd)	
CB	CB	Clear bit	Clear bit n (CB __ TamNonReg menu)	Function (monadic)	
ceil	ceil	Ceiling	Ceiling (type real)	Function (monadic)	
CENTRL	CENTRL	(Plot) Central	Orthogonal curve fit and show parameters, requires 30 data pairs (Opens the graph mode and (re)starts menu SCATR)	Command (special)	
CF	CF	Clear flag	Clear flag (CF __ TamFlag menu)	Command	
CHS	+/-	Change sign	Change sign	Function (monadic)	
CLALL	CLALL	Clear all	Clear all (programs and data)	Command	
CLBKUP	CLBKUP	Clear (delete) Backup	Delete configuration backup file (LOAD ; SAVE)	Command	
CLCVAR	CLCVAR	Clear current (program) variables	Clear current program variables	Command	
CLFALL	CLFALL	Clear (user) flags	Clear all user flags	Command	
CLGRF	CLGRF	Clear graph	Clear graph	Command	
CLK12 ()	CLK12	12 hour clock	12 hour time display	Setting (pgm)	OFF
CLK24 (+)	CLK24	24 hour clock	24 hour time display	Setting (pgm)	ON
CLLCD	CLLCD	Clear LCD (screen)	Clear (part of) the display, depending on X and Y	Command	
CLMENU	CLMENU	Clear (program) menu	Clear the programmable menu	Command	
CLP	CLP	Clear (current) program	Clear current program ; careful! (CLP _ TamLb(Alpha) menu)	Command	
CLPALL	CLPALL	Clear all programs	Clear all programs	Command	
CLREGS	CLREGS	Clear all registers	Clear all registers	Command	
CLRMOD	CLRMOD	Clear modes	Clear all menus, entry, fraction and base modes (Longerpress (also in AIM) [EXIT])	Command	
CLSTK	CLSTK	Clear stack	Clear all stack data (Longpress [⇐])	Command	
CLX	CLX	Clear X	Clear X-register	Command	
CLΣ	CLΣ	Clear statistics	Clear all statistics data (and delete STATS and HISTO matrices)	Command	
CNST	CNST	Constant	Get constant (0..78) showing temporary information (abbreviation and symbol) (CNST ___ TamNonReg menu)	Command	
cn(u,m)	cn(u,m)	Elliptic cosine	Elliptic cosine (u = X ; m = Y)	Function (dyadic)	
COMB	C _{y,x}	Combinations	Combinations of X out of Y	Function (dyadic)	
COMPLEX	COMPLEX	Complex	Convert to or from complex number (a ENTER b COMPLEX returns a+bi or a∠b (using b angle tag or ADM) ; COMPLEX returns Y : a, X : b)	Function (dyadic)	
conj	conj	Conjugate	Conjugate	Function (monadic)	
CONVG?	CONVG?	Convergence?	Tests convergence of X and Y using binary coded parameter for comparison mode and special numbers ; tolerance is derived from setting SDIGS (CONVG? _ TamNonReg menu)	Function (dyadic)	
CORR	r	Correlation	Correlation	Command	
COS	COS	Cosine	Cosine	Function (monadic)	
cosh	cosh	Hyperbolic cosine	Hyperbolic cosine	Function (monadic)	
COV	cov	Covariance	Covariance	Command	
CPX <i>i</i> (+)	CPX <i>i</i>	Imaginary i	Set for the letter i representing the imaginary number (<i>i</i> or <i>j</i> displayed in stack and on soft buttons)	Setting (pgm)	ON
CPX <i>j</i> ()	CPX <i>j</i>	Imaginary j	Set for the letter j representing the imaginary number (<i>i</i> or <i>j</i> displayed in stack and on soft buttons)	Setting (pgm)	OFF
CPXmul []	CPXmul	Complex multiplier	Complex multiplier (If clear, trailing i or j is used)	Setting	OFF
CPXRES [+]	CPXRES	Complex results	Set to allow complex results for real input ; auto-set when entering complex input (FF I ; Info : SBI depends on SBcr)	Setting	ON
CPXRES0	CPXRES0	Real results	Do not allow complex results for real input ; an error will occur for such events (For programming purposes)	Command	
CPXRES1	CPXRES1	Complex results	Allow complex results for real input ; an error will not occur for such events (For programming purposes)	Command	
CPX?	CPX?	Complex?	Test X is complex	Function (monadic)	

Catalog	Label	FullName	Description	Type	Default
CROSS	cross	Cross (x)	Cross product (215)	Function (dyadic)	
CX→RE	CX→RE	Complex to real	Convert complex to reals (in POLAR, using angle tag or ADM) (Re = ; Im = or r = ; θ = (2 stack levels))	Function (monadic)	
DATE	DATE	Date	Current date (Weekday)	Command	
DATE→	DATE→	Date to stack	Convert date to day, month, year in stack according to DISP or CLK settings for date format	Function (monadic)	
DAY	DAY	Day	Day (of date)	Command	
DBLx	DBLx	DBLR	Double word length remainder	Function (dyadic)	
DBL*	DBL*	Double multiply	Double word length multiply (result in Y - least and X - most significant digit)	Function (dyadic)	
DBL/	DBL/	Double divide	Double word length divide (Z - least, Y - most significant digits) / (X)	Function (triadic)	
DEC	DEC	Decimal	Convert X to decimal and toggle decimal mode	Setting (pgm)	OFF
DECOMP	DECOMP	Decompose	Converts (improper) fraction to nominator in Y, denominator in X (Honours settings DENANY and DENFIX)	Function (monadic)	
DECR	DEC	Decrement	Decrement by 1 (DEC __ Tam menu)	Command	
DEG (+)	DEG	Set ADM to DEG	Set ADM to degrees mode (SBI depends on SBang)	Setting (pgm)	ON
DELITM	DELITM	Delete items	Delete user defined items, selected from category (programs, variables, menus) (DELITM_menu DELITM)	MENU (item)	
DENANY []	DENANY	Denominator any	Any denominator will be used (fraction mode) (Result depends on setting DMX ; SBI depends on SBfrac)	Setting	OFF
DENFIX []	DENFIX	Denominator fixed	Only specified denominator will be used (fraction mode) (SBI depends on SBfrac)	Setting	OFF
DET	DET	Determinant	Determinant	Function (legacy)	
DISK?	DISK?	Disk state	Status of the FAT disk (Disk)	Command	
DMCP	DMCP	Access DMCP menu	Access DMCP menu (use EXIT to return without reset ; hardware only) (Does not autosave backup file to FAT)	Command	
DMX ₆₄	DMX	Denominator maximum	Fractions: D.MAX set to nn ; decimal fraction, default shown as 64ths (DMX ____ TamNonReg menu)	Setting (pgm)	64
DMY ()	DMY	Day month year	Date display mode DD.MM.YYYY (DD.MM.YYYY)	Setting	OFF
dn(u,m)	dn(u,m)	Elliptic delta amplitudinis	Elliptic delta amplitudinis (u = X ; m = Y)	Function (dyadic)	
DOT	dot	Dot (•)	Dot product (8729)	Function (dyadic)	
DRG	DRG	Degrees-radians-gradians	Add ADM tag to untagged value in X, convert tagged value to degrees, next radians, next gradians (cyclic) (Setting tag for complex X also sets POLAR mode for X according to ADM)	Function (cyclic ; nonpgm ; m)	
DROP	DROP+	Drop	Drop one stack level (Double [⇐])	Command	
DROPy	DROPy	Drop Y	Drop Y from stack	Command	
DSE	DSE	Decrement skip on equal	Decrement skip on equal (DSE __ Tam menu)	Command	
DSL	DSL	Decrement skip on less	Decrement skip on less (DSL __ Tam menu)	Command	
DSP ₃	DSP	Display precision	Set display mode precision (only) (DSP __ TamNonReg menu)	Setting (pgm)	3
DSTACK _n	DSTACK	Display stack registers	Set the number of stack registers to be displayed (DSTACK _ TamNonReg menu)	Setting	4
DSZ	DSZ	Decrement skip on zero	Decrement skip on zero (DSZ __ Tam menu)	Command	
DT→J	DT→J	Date-time to julian day number	Convert date, time in stack to julian day number (JDN) (Date, time can be in Y, X or X, Y)	Function (dyadic)	
D,MS ()	D,MS	Set ADM to D,MS	Set ADM to sexagesimal degrees mode (SBI depends on SBang)	Setting (legacy)	OFF
D→J	D→J	Date to julian day number	Convert date to julian day number (JDN) (.0 equals noon!)	Function (monadic)	
EIGVAL	EIGVAL	Eigenvalue	Eigenvalue	Function (monadic)	
EIGVEC	EIGVEC	Eigenvector	Eigenvector	Function (monadic)	
e^{ix}	e^{ix}	Euler's formula	Rotate complex unit vector by X radians : $e^{ix} = \cos(x) + i \sin(x)$ (i or j displayed in stack and on soft buttons)	Function (monadic)	
END	END	End	End statement	Command (PEM)	
ENDP (•)	End	TVM end payments	Payments at the end of each period (time value of money) (SBI depends on SBTvm)	Setting (pgm)	ON
ENG ()	ENG	Engineering notation	Set numeric display mode to ENgineering notation with nn+1 digits (ENG __ TamNonReg menu)	Setting (pgm)	OFF
ENGOVR ()	ENGOVR	Engineering display large reals	Change display to ENG for reals too large to display in full (FF A)	Setting (pgm)	OFF
ENORM	ENORM	Euclidean norm	Euclidean norm	Function (monadic)	
ENTER↑	ENTER	Enter	Enter input value to X (optionally also to Y) or push/duplicate value already in X to Y	Command	
ENTRY?	ENTRY?	(Data) entry?	Test internal entry flag for data entry	Command	
EQ.DEL	DELETE	Delete	Delete equation	Command	
EQ.EDI	EDIT	Edit equation	Edit equation (previous equation loaded) (EIM = Equation Input Mode ; starts lowercase)	Command (submnu)	
EQ.NEW	NEW	Equation editor	Create new equation (previous equation pushed) (EIM = Equation Input Mode ; starts lowercase)	MENU (item)	
erf	erf	Error function	Error function	Function (monadic)	
erfc	erfc	Complementary error function	Complementary error function	Function (monadic)	
eRPN [•]	eRPN	Entry RPN	Entry RPN	Setting	ON
eRPN	eRPNon	Entry RPN on	Set stack to entry RPN mode (no DUP) (For programming purposes)	Setting (pgm)	
eRPN?	eRPN?	Entry RPN	Entry RPN mode : 1 ; classic RPN mode : 0 ; set by ON (For programming purposes)	Command	
ERR	ERR	Error	Raise error and show error message (ERR __ TamNonReg menu)	Command (PEM)	

Catalog	Label	FullName	Description	Type	Default
EVEN?	EVEN?	Even?	Test X is integer AND even	Function (monadic)	
e^x	e^x	e to the power x	Raise e to the power in the X-register	Function (monadic)	
EXFRAC []	EXFRAC	Extended fractions	Extended fractions, helper mode to find approximate multiples or fractions of the irrational numbers $\sqrt{2}$, $\sqrt{3}$, $\sqrt{5}$, ϕ , π and e (\approx <approximation> ; SBI depends on SBfrac)	Setting	OFF
EXITALL	EXITall	Exit all	Exit all menus ; return to MyMenu (interactive mode) ; exit VarMNU (program mode)	Command	
ExpF []	ExpF	Exponential fit	Exponential curve fitting	Setting	OFF
Expon _p	Expon _p	Exponential pdf	Exponential probability density function	Function (tbd)	
Expon _l	Expon _l	Exponential cdf (lower)	Exponential cumulative distribution (lower tail)	Function (tbd)	
Expon _u	Expon _u	Exponential cdf (upper)	Exponential cumulative distribution (upper tail)	Function (tbd)	
Expon ⁻¹	Expon ⁻¹	Exponential (inverse)	Exponential probability inverse function	Function (tbd)	
EXPT	EXPT	Exponent	Exponent of number in the X-register	Function (monadic)	
e^{x-1}	e^{x-1}	e^{x-1}	More accurate calculation of e^{x-1} for $x \approx 0$	Function (monadic)	
E(m)	E(m)	Complete elliptic integral (2nd)	Complete elliptic integral of the 2nd kind (m = X)	Function (monadic)	
E(ϕ ,m)	E(ϕ ,m)	Incomplete elliptic integral (2nd)	Incomplete elliptic integral of the 2nd kind ($\phi = X$; m = Y)	Function (dyadic)	
FB	FB	Flip bit	Flip bit n (FB __ TamNonReg menu)	Function (monadic)	
FBR	FBR	Font browser	Browse system fonts (character tables)	Browser	
FCOM, ()	COM,	FP separator comma	Set fractional part separator to comma	Setting (pgm)	OFF
FC?	FC?	Flag clear?	Test flag clear? (FC? __ TamFlag menu)	Command	
FC?C	FC?C	Flag clear? and clear	Test flag clear? and clear (FC?C __ TamFlag menu)	Command	
FC?F	FC?F	Flag clear? and flip	Test flag clear? and flip (FC?F __ TamFlag menu)	Command	
FC?S	FC?S	Flag clear? and set	Test flag clear? and set (FC?S __ TamFlag menu)	Command	
FDOT· ()	DOT·	FP separator dot	Set fractional part separator to dot	Setting (pgm)	OFF
FF	FF	Flip flag	Flip flag (toggle) (FF __ TamFlag menu)	Command	
fg.FUL (*)	fg.FUL	Menu fg-highlighting Full	Show full horizontal f- and g-lines indicating the state of the f- or g-shift in menu	Setting	ON
fg.LIM ()	fg.LIM	Menu fg-highlighting Limited	Show limited horizontal f- and g-lines indicating the state of the f- or g-shift in menu	Setting	OFF
fg.OFF ()	fg.OFF	Menu fg-highlighting Off	Do not show horizontal f- and g-lines indicating the state of the f- or g-shift in menu	Setting	OFF
FIB	FIB	Fibonacci	Fibonacci number n, where n = X	Function (monadic)	
FILL	FILL	Fill stack	Fill stack with value in the X-register	Command	
FIX ()	FIX	Fixed notation	Set numeric display mode to FIXed notation with nn+1 digits (FIX __ TamNonReg menu)	Setting (pgm)	OFF
FLGS	FLGS	Flag browser	Show all flags on one page (0 = clear, 1 = set) ; show status page(s) on Up/Dn (Compare FLAGS.STATUS)	Browser	
floor	floor	Floor	Floor (type real)	Function (monadic)	
FNONE ()	NONE	No FP separator	No fractional part separator (Menu shows symbol ø)	Setting (pgm)	OFF
FNSPC· ()	NSPC·	FP separator narrow space	Set fractional part separator to narrow space	Setting (pgm)	OFF
FP	FP	Fractional part	Fractional part (#F (closed number))	Function (monadic)	
FPER. ()	PER.	FP separator period	Set fractional part separator to period	Setting (pgm)	OFF
FPGRP ₃	FPGRP	FP group size	Set fractional part group size (2..9) (FPGRP __ TamNonReg menu)	Setting (pgm)	3
$F_p(x)$	$F_p(x)$	Fisher's F pdf	Fisher's F probability density function	Function (monadic)	
FP?	FP?	Fractional part?	Test X has nonzero fractional part	Function (monadic)	
FRCSRN []	FRCSRN	Fraction show register name	Fractions are shown with register names (x, y, z, t < or = or > or \approx)	Setting	OFF
FSPC· (*)	SPC·	FP separator space	Set fractional part separator to space	Setting (pgm)	ON
FS?	FS?	Flag set?	Test flag set? (FS? __ TamFlag menu)	Command	
FS?C	FS?C	Flag set? and clear	Test flag set? and clear (FS?C __ TamFlag menu)	Command	
FS?F	FS?F	Flag set? and flip	Test flag set? and flip (FS?F __ TamFlag menu)	Command	
FS?S	FS?S	Flag set? and set	Test flag set? and set (FS?S __ TamFlag menu)	Command	
FTICK' ()	TICK'	FP separator narrow tick	Set fractional part separator to narrow tick	Setting (pgm)	OFF
FUNDR ()	UNDR	FP separator underscore	Set fractional part separator to underscore	Setting (pgm)	OFF
FWCOM, ()	WCOM,	FP separator wide comma	Set fractional part separator to wide comma	Setting (pgm)	OFF
FWDOT· ()	WDOT·	FP separator wide dot	Set fractional part separator to wide dot	Setting (pgm)	OFF
FWPER. ()	WPER.	FP separator wide period	Set fractional part separator to wide period	Setting (pgm)	OFF
FWSPC·· ()	WSPC··	FP separator double space	Set fractional part separator to double space	Setting (pgm)	OFF
FWTICK' ()	WTICK'	FP separator tick	Set fractional part separator to tick	Setting (pgm)	OFF
F(ϕ ,m)	F(ϕ ,m)	Incomplete elliptic integral (1st)	Incomplete elliptic integral of the 1st kind ($\phi = X$; m = Y)	Function (dyadic)	
$F_l(x)$	$F_l(x)$	Fisher's F cdf (lower)	Fisher's F cumulative distribution (lower tail)	Function (monadic)	
$F_u(x)$	$F_u(x)$	Fisher's F cdf (upper)	Fisher's F cumulative distribution (upper tail)	Function (monadic)	
$F^{-1}(p)$	$F^{-1}(p)$	Fisher's F (inverse)	Fisher's F probability inverse function	Function (tbd)	
$f'(x)$	$f'(x)$	$f'(x)$	First derivative of f at x ($f'(x)$ __ TamLbl(Alpha) menu)	Function (monadic)	

Catalog	Label	FullName	Description	Type	Default
$f''(x)$	$f''(x)$	$f''(x)$	Second derivative of f at x ($f''(x)$ __ TamLb(Alpha) menu)	Function (monadic)	
GaussF []	GaussF	Gauss fit	Gauss curve fitting	Setting	OFF
GCD	GCD	Greatest common divisor	Greatest common divisor of X and Y	Function (dyadic)	
g_d	g_d	g_d	Gudermannian function	Function (monadic)	
g_d^{-1}	g_d^{-1}	g_d^{-1}	Inverse Gudermannian function	Function (monadic)	
Geom _p	Geom _p	Geometric pdf	Geometric probability density function	Function (tbd)	
Geom _▲	Geom _▲	Geometric cdf (lower)	Geometric cumulative distribution (lower tail)	Function (tbd)	
Geom _▲	Geom _▲	Geometric cdf (upper)	Geometric cumulative distribution (upper tail)	Function (tbd)	
Geom ⁻¹	Geom ⁻¹	Geometric (inverse)	Geometric probability inverse function	Function (tbd)	
GRAD ()	GRAD	Set ADM to GRAD	Set ADM to gradients mode (SBI depends on SBang)	Setting (pgm)	OFF
GTO	GTO	Go to	Go to local/global label or line (GTO __ TamLb(Alpha) menu)	Command	
GTO.	GTO.	Go to label or step	Go to label or step (GTO. ____ GTO menu)	Command	
g.2Tp [*]	g.2Tp	Function keys g-shortcut	Allow double tapping the FN-keys for a g-function (Blocked for (navigation) arrows in editors)	Setting	ON
HEX	HEX	Hexadecimal	Convert X to hexadecimal and toggle hexadecimal mode	Setting (pgm)	OFF
HIDE ₀	HIDE	Hide small values	Hide (i.e. display '0.' instead of) all real numbers or parts with absolute values $< 10^{-n}$ with $n = IP(x)$, $12 \leq n \leq 99$; useful e.g. in matrices; reset by HIDE 0 (HIDE __ TamNonReg menu)	Setting (pgm)	0
HIDE?	HIDE?	Get hide (setting)	Current setting of HIDE	Command	
HISTOX	HISTOX	Histogram X	Evaluate first column of STATS and store in HISTO (nBINS : ; \downarrow BIN : ; \uparrow BIN : (3 stack levels))	Command	
HISTOY	HISTOY	Histogram Y	Evaluate second column of STATS and store in HISTO (nBINS : ; \downarrow BIN : ; \uparrow BIN : (3 stack levels))	Command	
H_n	H_n	Hermite polynomials (probability)	Hermite polynomials (probability) ($x = X$; $n = Y$)	Function (dyadic)	
HNORM	HNORM	Histogram Normal	Fit Gauss distribution through HISTO data	Command	
H_{np}	H_{np}	Hermite polynomials (physics)	Hermite polynomials (physics) ($x = X$; $n = Y$)	Function (dyadic)	
HOME.3 [*]	HOME.3	HOME menu fff shortcut	HOME menu activated by triple shift	Setting	ON
HOUR	HOUR	Hours	Hours (of time)	Function (monadic)	
HPLLOT	HPLLOT	Histogram	Histogram plotting (HPLLOT)	MENU (item)	
Hyper _p	Hyper _p	Hypergeometric pdf	Hypergeometric probability density function	Function (tbd)	
Hyper _▲	Hyper _▲	Hypergeometric cdf (lower)	Hypergeometric cumulative distribution (lower tail)	Function (tbd)	
Hyper _▲	Hyper _▲	Hypergeometric cdf (upper)	Hypergeometric cumulative distribution (upper tail)	Function (tbd)	
Hyper ⁻¹	Hyper ⁻¹	Hypergeometric (inverse)	Hypergeometric probability inverse function	Function (tbd)	
HypF []	HypF	Hyperbolic fit	Hyperbolic curve fitting	Setting	OFF
ICOM, ()	COM,	IP separator comma	Set integer part separator to comma	Setting (pgm)	OFF
IDIV	IDIV	Integer divide	Integer divide	Function (dyadic)	
IDIVR	IDIVR	Integer divide and remainder	Integer divide (X) and remainder (Y)	Function (dyadic)	
IDOT. ()	DOT.	IP separator dot	Set integer part separator to dot	Setting (pgm)	OFF
Im	Im	Imaginary part	Imaginary part of complex number	Function (monadic)	
INC	INC	Increment	Increment by 1 (INC __ Tam menu)	Command	
INDEX	INDEX	Index the matrix	Index the matrix (INDEX __ Tam menu)	Command	
INONE ()	NONE	No IP separator	No integer part separator (Menu shows symbol \emptyset)	Setting (pgm)	OFF
INPUT	INPUT	Input	Halt program execution, push current value and accept input for variable or register (INPUT __ Tam menu)	Command (PEM)	
INSPC. ()	NSPC.	IP separator narrow space	Set integer part separator to narrow space	Setting (pgm)	OFF
INT?	INT?	Integer?	Test X has zero fractional part	Function (monadic)	
INVRT	INVRT	Invert matrix	Inverse of matrix	Function (legacy, monadic)	
IP	IP	Integer part	Integer part (type real) (#I (closed number))	Function (monadic)	
IPER. ()	PER.	IP separator period	Set integer part separator to period	Setting (pgm)	OFF
IPGRP ₃	IPGRP	IP group size	Set integer part group size (2..9); not for SCI or ENG notation (IPGRP __ TamNonReg menu)	Setting (pgm)	3
IPGRP ₁₀	IPGRP1	IP first group size	Set integer part first group size; parameter 0 means follow IPGRP (IPGRP1 __ TamNonReg menu)	Setting (pgm)	0
IPGRP _{1x0}	IPGRP1x	IP first group extension	Extend first group to allow one additional digit up to maximum specified by parameter (IPGRP1x __ TamNonReg menu)	Setting (pgm)	0
ISE	ISE	Increment skip on equal	Increment skip on equal (ISE __ Tam menu)	Command	
ISG	ISG	Increment skip on greater	Increment skip on greater (ISG __ Tam menu)	Command	
ISM?	ISM?	Integer sign mode	Sign mode for short integers; set by UNSIGN; SIGNMT	Command	
ISPC. (*)	SPC.	IP separator space	Set integer part separator to space	Setting (pgm)	ON
ISZ	ISZ	Increment skip on zero	Increment skip on zero (ISZ __ Tam menu)	Command	
ITICK' ()	TICK'	IP separator narrow tick	Set integer part separator to narrow tick	Setting (pgm)	OFF

Catalog	Label	FullName	Description	Type	Default
IUNDR_ ()	UNDR_	IP separator underscore	Set integer part separator to underscore	Setting (pgm)	OFF
IWCOM, ()	WCOM,	IP separator wide comma	Set integer part separator to wide comma	Setting (pgm)	OFF
IWDOT* ()	WDOT*	IP separator wide dot	Set integer part separator to wide dot	Setting (pgm)	OFF
IWPER. ()	WPER.	IP separator wide period	Set integer part separator to wide period	Setting (pgm)	OFF
IWSPC... ()	WSPC...	IP separator double space	Set integer part separator to double space	Setting (pgm)	OFF
IWTICK' ()	WTICK'	IP separator tick	Set integer part separator to tick	Setting (pgm)	OFF
I_{xyz}	I_{xyz}	I_{xyz}	Regularised (incomplete) Beta function	Function (triadic)	
Γ_p	Γ_p	Γ_p	Regularised Gamma function (P)	Function (dyadic)	
Γ_q	Γ_q	Γ_q	Regularised Gamma function (Q)	Function (dyadic)	
$i\Pi_n$	$i\Pi_n$	Integer product (programmable)	Integer product using specified program, with iteration counter, interrupt by keypress ($i\Pi_n$ __ TamLbl(Alpha) menu ; <from> ENTER <to> ENTER <step>)	Command	
$i\Sigma_n$	$i\Sigma_n$	Integer sum (programmable)	Integer sum using specified program, with iteration counter, interrupt by keypress ($i\Sigma_n$ __ TamLbl(Alpha) menu ; <from> ENTER <to> ENTER <step>)	Command	
I+	I+	Increment row index	Increment row index	Command	
I-	I-	Decrement row index	Decrement row index	Command	
JG.1582	JG.1582	Julian-Gregorian transition 1582	Set Julian-Gregorian transition date to 1582-10-15 (First Gregorian day set: 1582-10-15)	Setting (pgm)	
JG.1752	JG.1752	Julian-Gregorian transition 1752	Set Julian-Gregorian transition date to 1752-09-14 (First Gregorian day set: 1752-09-14)	Setting (pgm)	
JG.1873	JG.1873	Julian-Gregorian transition 1873	Set Julian-Gregorian transition date to 1873-01-01 (First Gregorian day set: 1873-01-01)	Setting (pgm)	
JG.1949	JG.1949	Julian-Gregorian transition 1949	Set Julian-Gregorian transition date to 1949-10-01 (First Gregorian day set: 1949-10-01)	Setting (pgm)	
$J_y(x)$	$J_y(x)$	$J_y(x)$	Bessel function of the 1st kind and order y	Function (monadic)	
J+	J+	Increment column index	Increment column index	Command	
J-	J-	Decrement column index	Decrement column index	Command	
J/G	J/G	Set Julian-Gregorian (transition)	Set the day that Julian date changes over to Gregorian date (using date in X) (Input real number according to function x→DATE)	Function (monadic)	1752-09-14
J/G?	J/G?	Get Julian-Gregorian (transition)	Get the day that Julian date changes over to Gregorian date (Set by J/G)	Command	
J→DT	J→DT	Julian day number to date-time	Convert julian day number (JDN) to date, time in stack	Command	
KEY	KEY	Key	Used in programs to form program step for KEYG → KEY nn GTO __ or KEYX → KEY nn XEQ __ (KEY __ TamNonReg menu)	Command (PEM)	
KEY _{A-F} [*]	KEY _{A-F}	A-F keys	Set row 2 keys to be used for entry of digits A-F while in #BASE mode	Setting	ON
KEYG	KEYG	Key go to	Used in programs to specify label to go to when soft button (1-18), [▲] (19), [▼] (20) or EXIT (21) is pressed (KEY __ TamKey menu ; KEY nn GTO __ TamLbl(Alpha) menu)	Command (PEM)	
KEYX	KEYX	Key execute	Used in programs to specify label to execute when soft button (1-18), [▲] (19), [▼] (20) or EXIT (21) is pressed (KEY __ TamKey menu ; KEY nn XEQ __ TamLbl(Alpha) menu)	Command (PEM)	
KEY?	KEY?	Key pressed?	Test key was pressed (store keycode in register) (KEY? __ Tam menu)	Command	
KTYP?	KTYP?	Key type	Key type for key code returned by KEY? (KTYP? __ Tam menu)	Function (monadic)	
K(m)	K(m)	Complete elliptic integral (1st)	Complete elliptic integral of the 1st kind (m = X)	Function (monadic)	
LASTx	LASTx	Last X	Recall last X (register L)	Command	
LB	LB	Binary logarithm	Binary logarithm (base 2)	Function (monadic)	
LBL	LBL	Label	Create local/global label (LBL __ TamLbl(Alpha) menu)	Command (PEM)	
LBL?	LBL?	Label exists?	Test label exists (LBL? __ TamLbl(Alpha) menu)	Command	
LCM	LCM	Least common multiple	Least common multiple of X and Y	Function (dyadic)	
LEAD.0 []	LEAD.0	Leading zeros	Leading zeros are on (shortint bases 2, 4, 8, 16) (FF L)	Setting	OFF
LEAP?	LEAP?	Leap year?	Test date is in leap year	Function (monadic)	
LgNrm _p	LgNrm _p	Log normal pdf	Log normal probability density function	Function (tbd)	
LgNrm _l	LgNrm _l	Log normal cdf (lower)	Log normal cumulative distribution (lower tail)	Function (tbd)	
LgNrm _u	LgNrm _u	Log normal cdf (upper)	Log normal cumulative distribution (upper tail)	Function (tbd)	
LgNrm ⁻¹	LgNrm ⁻¹	Log normal (inverse)	Log normal probability inverse function	Function (tbd)	
LinF [*]	LinF	Linear fit	Linear curve fitting	Setting	ON
LINPOL	LINPOL	Linear interpolation	Linear interpolation ; also works for complex numbers (<from> ENTER <towards> ENTER <at> (fraction))	Function (triadic)	
LINT	LINT	Long integer	Convert to long integer (max 1000 digits) (#ENTER)	Function (monadic)	
LISTXY	LISTXY	List graph coordinates	List the actual STATS graph coordinates (7 digit floating point)	Command (deprecated)	
LJ	LJ	Left justify	Left justify (within word size) ; returns shift in X and result in Y	Function (monadic)	
L_m	L_m	Laguerre polynomials	Laguerre polynomials (x = X ; m = Y)	Function (dyadic)	
$L_{m\alpha}$	$L_{m\alpha}$	Laguerre generalised polynomials	Laguerre generalised polynomials (x = X ; m = Y ; $\alpha = Z$)	Function (triadic)	
LN	LN	Natural logarithm	Natural logarithm (base e)	Function (monadic)	
LNβ	LNβ	LNβ	Natural logarithm of Euler's Beta function	Function (dyadic)	

Catalog	Label	FullName	Description	Type	Default
LNΓ	LNΓ	LNΓ	Natural logarithm of the Gamma function	Function (monadic)	
LN(1+x)	LN(1+x)	LN(1+x)	More accurate calculation of LN(1+x) for $x \approx 0$	Function (monadic)	
LOAD	LOAD	Load full backup	Load full backup from SAVE file C47.sav in FAT and auto-clear user mode (SAVFILES ; inverse : SAVE)	Command (nonpgm)	
LOADP	LOADP	Load program	Load program from SAVE file C47.sav in FAT (SAVFILES ; inverse : SAVE)	Command (nonpgm)	
LOADR	LOADR	Load registers	Load registers from SAVE file C47.sav in FAT (SAVFILES ; inverse : SAVE)	Command (nonpgm)	
LOADSS	LOADSS	Load system state	Load system state from SAVE file C47.sav in FAT (SAVFILES ; inverse : SAVE)	Command (nonpgm)	
LOADST	LOADST	Load state file	Load state file from SAVE file <state.s47> in FAT (File open dialog (STATE) ; Inverse : SAVEST)	Command (nonpgm)	
LOADV	LOADV	Load variables	Load variables from SAVE file C47.sav in FAT (SAVFILES ; inverse : SAVE)	Command (nonpgm)	
LOADΣ	LOADΣ	Load Sigma registers	Load Sigma registers from SAVE file C47.sav in FAT (SAVFILES ; inverse : SAVE)	Command (nonpgm)	
LocR	LocR	Allocate local registers	Allocates n local registers and 16 local flags (current routine) (LocR __ TamNonReg menu)	Command (PEM)	
LocR?	LocR?	Number of local registers	Number of local registers (current routine) ; set by LocR	Command	
LOG	LOG	Common logarithm	Common logarithm (base 10)	Function (monadic)	
LogF []	LogF	Logarithmic fit	Logarithmic curve fitting	Setting	OFF
Logis _p	Logis _p	Logistic (inverse)	Logistic probability inverse function	Function (tbd)	
Logis _u	Logis _u	Logistic cdf (upper)	Logistic cumulative distribution (upper tail)	Function (tbd)	
Logis _d	Logis _d	Logistic pdf	Logistic probability density function	Function (tbd)	
Logis ⁻¹	Logis ⁻¹	Logistic cdf (lower)	Logistic cumulative distribution (lower tail)	Function (tbd)	
LOG _x y	LOG _x y	LOG _x y	Logarithm of Y for base X	Function (dyadic)	
LRG_LI [•]	LRG_LI	Large longint display	Large longint display	Setting	ON
L.R.	L.R.	Linear Regression	Linear Regression	Command	
MANT	MANT	Mantissa	Mantissa	Function (monadic)	
MASKL	MASKL	Left bit mask	Set left n bits to use as mask (MASKL __ TamNonReg menu)	Function (monadic)	
MASKR	MASKR	Right bit mask	Set right n bits to use as mask (MASKR __ TamNonReg menu)	Function (monadic)	
MATR?	MATR?	Matrix?	Test X is a matrix	Function (monadic)	
Mat_X	Mat X	Matrix X	Create reserved matrix variable Matrix X (Mat_X) and solve simultaneous equations	Command	
max	max	Maximum	Maximum of X and Y	Function (dyadic)	
MDY ()	MDY	Month day year	Date display mode MM/DD/YYYY (MM/DD/YYYY)	Setting (pgm)	OFF
MEM?	MEM?	Memory (RAM)	Amount of free RAM memory	Command	
MENU	MENU	Display (program) menu	Display the programmable menu	Command (PEM)	
MIN	MIN	Minutes	Minutes (of time)	Function (monadic)	
min	min	Minimum	Minimum of X and Y	Function (dyadic)	
MIRROR	MIRROR	Mirror bits	Flip bits	Function (monadic)	
MOD	MOD	Modulo	Y modulo X	Function (dyadic)	
MONTH	MONTH	Month	Month (of date)	Function (monadic)	
MSG	MSG	Message	Show error message (MSG __ Tam menu)	Command	
MULT× (•)	MULT×	Multiplication symbol ×	Multiplication symbol × for exponential (and for complex numbers if CPXmul is set)	Setting (pgm)	ON
MULT• ()	MULT•	Multiplication symbol •	Multiplication symbol • for exponential (and for complex numbers if CPXmul is set)	Setting (pgm)	OFF
MULT (•)	MULT	Set ADM to MULT	Set ADM to multiple of pi radians mode (SBI depends on SBAng)	Setting (kegcy)	OFF
MVAR	MVAR	Menu variable	Define menu variable for VarMNU (MVAR __ Tam menu)	Command (PEM)	
MyM [•]	MyM	MyMenu shown	Base MyMenu shown (when all menus are exited using EXIT)	Setting	ON
MYM.3 [•]	MYM.3	MyMenu fff shortcut	MyMenu activated by triple shift	Setting (strike)	OFF
M.DELR	DEL	Delete row	Delete row from matrix	Command	
M.DIM	DIM	Dimension	Dimension (M.DIM __ Tam menu)	Function (monadic)	
M.DIM?	DIM?	Matrix dimension (X)	Dimensions of matrix in X	Function (monadic)	
M.EDI	EDIT	Edit matrix (X)	Edit matrix (X-register) (MIM = Matrix Input Mode ; CAT.MENUS M.EDIT)	MENU (item)	
M.EDIN	EDITN	Edit matrix (named)	Edit matrix (named variable) (M.EDIN __ Tam menu)	Command (submnu)	
M.ENG	M.ENG	MyMenu ENG	Populate MyMenu with engineering functions (Documented as MyMenu page 1 (default))	Command	
M.FIN	M.FIN	MyMenu FIN	Populate MyMenu with financial functions (Documented as MyMenu page 2)	Command	
M.GET	GETM	Get submatrix	Get submatrix	Command	
M.GOTO	GOTO	Go to matrix element	Go to matrix element using GOTO Row ; GOTO Column (GOTO Row ; GOTO Column)	Command	
M.GROW ()	GROW	Grow (matrix edit)	Matrix edit in grow mode (SBI depends on SBmx)	Setting	OFF
M.INSR	INSR	Insert row	Insert row into matrix	Command	
M.LU	M.LU	LU decomposition	Decompose matrix into lower (L) in Y and upper (U or R) matrix in X ; pivot matrix in Z (LU (LR) factorisation)	Function (monadic)	
M.NEW	NEW	New matrix	Create new matrix (MIM = Matrix Input Mode)	Command (submnu)	
M.OLD	OLD	Old (matrix)	Revert to old element (while editing)	Command	

Catalog	Label	FullName	Description	Type	Default
M.PUT	PUTM	Put submatrix	Put submatrix	Command	
M.QR	M.QR	QR decomposition	Decompose matrix into unitary (Q) in Y and upper (R) matrix in X (QR factorisation)	Function (monadic)	
M.R↔R	R↔R	Swap rows	Swap rows	Command	
M.SQR?	M.SQR?	Square matrix?	Test matrix is square	Function (monadic)	
M.WRAP (+)	WRAP	Wrap (matrix edit)	Matrix edit in wrapping mode (SBI depends on SBmx)	Setting	ON
M→zyx	M→zyx	Decompose 3x1 matrix	Decompose 3x1 matrix to ZYX (zyx→M)	Function (monadic)	
NAND	NAND	Not AND	Logical not AND (bitwise)	Function (dyadic)	
NaN?	NaN?	Not a number?	Test X is Not-a-Number	Function (monadic)	
NBin _p	NBin _p	Negative binomial pdf	Negative binomial probability density function	Function (tbd)	
NBin _l	NBin _l	Negative binomial cdf (lower)	Negative binomial cumulative distribution (lower tail)	Function (tbd)	
NBin _u	NBin _u	Negative binomial cdf (upper)	Negative binomial cumulative distribution (upper tail)	Function (tbd)	
NBin ⁻¹	NBin ⁻¹	Negative binomial (inverse)	Negative binomial probability inverse function	Function (tbd)	
NEIGHB	NEIGHB	Neighbour	Neighbour value of X compared to Y (next integer or next machine representable real)	Function (dyadic)	
NEXTP	NEXTP	Next prime	Next prime number	Function (monadic)	
NOP	NOP	No operation	No operation (empty step)	Command (PEM ; deprecated)	
NOR	NOR	NOR	Logical not OR (bitwise)	Function (dyadic)	
Norm _p	Norm _p	Normal pdf	Normal probability density function	Function (tbd)	
Norm _l	Norm _l	Normal cdf (lower)	Normal cumulative distribution (lower tail)	Function (tbd)	
Norm _u	Norm _u	Normal cdf (upper)	Normal cumulative distribution (upper tail)	Function (tbd)	
Norm ⁻¹	Norm ⁻¹	Normal (inverse)	Normal probability inverse function	Function (tbd)	
NOT	NOT	NOT	Logical NOT	Function (monadic)	
nΣ	n	n	Number of samples	Command	
OCT	OCT	Octal	Convert X to octal and toggle octal mode	Setting (pgm)	OFF
ODD?	ODD?	Odd?	Test X is integer AND odd	Function (monadic)	
OFF	OFF	Off	Turn off calculator	Command	
op_A	[A]	Create 3x3 A-Matrix	Create 3x3 A-matrix relating to Fortescue's Symmetrical Components	Command	
op_a	a	Operator a	Insert value of $1 \angle 120^\circ$	Command	
op_a ²	a ²	Operator a ²	Insert value of $1 \angle 240^\circ$	Command	
op_i	i	Imaginary number	Complex number i ; displayed according to flag CPX,j (default: i) (In NIM, works like CC ; RECT input assumed always)	Command	
OR	OR	OR	Logical OR	Function (dyadic)	
OrthoF []	OrthoF	Orthogonal fit	Orthogonal curve fitting	Setting	OFF
ParabF []	ParabF	Parabolic fit	Parabolic curve fitting	Setting	OFF
PAUSE	PAUSE	Pause	Pause program for n ticks (0-99 ; one tick is 10 ms) ; continues after delay or on keypress (PAUSE __ TamNonReg menu)	Command (PEM)	
PERM	P _{yx}	Permutations	Permutations of X out of Y	Function (dyadic)	
PFX.All [+]	PFX.All	Show all prefixes	Show all SI unit prefixes in UNIT (prefix) display mode: 10^{-30} to 10^{30} ; if OFF, the range is 10^{-15} to 10^{15}	Setting	ON
PGMINT	PGMINT	Program for integrator	The program to be used by the integrator (PGMINT __ TamLbl(Alpha) menu)	Command	
PGMSLV	PGMSLV	Program for solver	The program to be used by the solver (PGMSLV __ TamLbl(Alpha) menu)	Command	
PIXEL	PIXEL	Display pixel	Display one pixel (X, Y)	Command	
PLAY	PLAY	Play sounds	Play sounds (input from nx3 or nx2 matrix variable having rows : [frequency, duration, volume] with frequency in Hz (0 = silent) ; duration in ms (max 2000) ; volume) ; volume element is optional (PLAY _ TamLabel menu)	Command	
PLSTAT	PLSTAT	Plot statistics	Plot statistics (PLOTMMU)	MENU (item)	
PLTRST	PLTRST	Reset plot	PLSTAT only: reset all plot options and redraw graph (Options include boxes, crosses, lines, scales)	Command	
P _n	P _n	Legendre polynomials	Legendre polynomials ($x = X$; $n = Y$)	Function (dyadic)	
POINT	POINT	Display point	Display 9 (3x3) pixels (X, Y)	Command	
Poiss _p	Poiss _p	Poisson (inverse)	Poisson probability inverse function	Function (tbd)	
Poiss _u	Poiss _u	Poisson cdf (upper)	Poisson cumulative distribution (upper tail)	Function (tbd)	
Poiss _l	Poiss _l	Poisson pdf	Poisson probability density function	Function (tbd)	
POLAR ()	POLAR	Polar	Polar representation of complex numbers (internal value is RECT) (FF X (TAM) ; Info : SBI depends on SBcpX)	Setting (pgm)	OFF
Poiss ⁻¹	Poiss ⁻¹	Poisson cdf (lower)	Poisson cumulative distribution (lower tail)	Function (tbd)	
PopLR	PopLR	Pop local registers	Pop local registers (no return to calling routine)	Command (PEM)	
PowerF []	PowerF	Power fit	Power curve fitting	Setting	OFF
PRIME?	PRIME?	Prime?	Test absolute value of integer part of X is prime	Function (monadic)	

Catalog	Label	FullName	Description	Type	Default
PRN	PRN	Print	Print	Command	
PRNTR []	PRNTR	Printer on/off	Set printer on/off	Setting	OFF
PROFF	PROFF	Printer off	Set printer off	Command	
PRON	PRON	Printer on	Set printer on	Command	
PUTK	PUTK	Put keycode in buffer	Copy keycode from register to keyboard buffer for immediate execution (PUTK __ Tam menu)	Command	
RAD ()	RAD	Set ADM to RAD	Set ADM to radians mode (SBI depends on SBang)	Setting (pgm)	OFF
RANGE?	RANGE?	Get range (setting)	Maximum number exponent (range) ; set by RNG	Command	
RANI#	RANI#	Random integer	Random integer : lower ENTER upper ; keeping input on stack (X: result, Y: upper, Z: lower)	Function (dyadic)	
RAN#	RAN#	Random number	Random number (real)	Command	
RCL	RCL	Recall (register)	Recall value from register or variable can be followed by +, -, *, ÷ for recall and add, recall and subtract, recall and multiply, recall and divide functions (RCL __ TamStoRcl(Alpha) menu)	Command	
RCL 3I	RCL 3I	Recall triple I	Copy R96, R97, R98 to X, Y, Z	Command	
RCL 3V	RCL 3V	Recall triple V	Copy R93, R94, R95 to X, Y, Z	Command	
RCL 3Z	RCL 3Z	Recall triple Z	Copy R90, R91, R92 to X, Y, Z	Command	
RCLCFG	RCLCFG	Recall configuration	Recall configuration from register or variable (RCLCFG __ TamStoRcl(Alpha) menu)	Command	
RCLEL	RCLEL	Recall current element	Recall current element	Command	
RCLIJ	RCLIJ	Recall current index	Recall current index	Command	
RCLS	RCLS	Recall stack	Recall complete stack from 4 or 8 registers (RCLS __ TamStoRcl(Alpha) menu)	Command	
RCL+	RCL+	Recall and add	Recall register or variable and add X (RCL+ __ TamStoRcl(Alpha) menu)	Function (monadic)	
RCL-	RCL-	Recall and subtract	Recall register or variable and subtract X (RCL- __ TamStoRcl(Alpha) menu)	Function (monadic)	
RCL*	RCL*	Recall and multiply	Recall register or variable and multiply by X (RCL* __ TamStoRcl(Alpha) menu)	Function (monadic)	
RCL/	RCL/	Recall and divide	Recall register or variable and divide by X (RCL/ __ TamStoRcl(Alpha) menu)	Function (monadic)	
RCL^	RCL^	Recall maximum	Recall maximum of X and register or variable (RCL^ __ TamStoRcl(Alpha) menu)	Command	
RCLv	RCLv	Recall minimum	Recall minimum of X and register or variable (RCLv __ TamStoRcl(Alpha) menu)	Command	
RCOM, ()	COM,	Radix comma	Radix decimal comma	Setting (pgm)	OFF
RDOT· ()	DOT·	Radix dot	Radix decimal dot	Setting (pgm)	OFF
RDP	RDP	Round to decimal places	Rounds to n decimal places (RDP __ TamNonReg menu)	Function (monadic)	
Re	Re	Real part	Real part of complex number	Function (monadic)	
READP	READP	Read program	Read program from WRITEP file <program.p47> in FAT (File open dialog (PROGRAMS) ; Inverse : WRITEP)	Command	
REAL?	REAL?	Real?	Test X is real	Function (monadic)	
RECT (+)	RECT	Rectangular	Rectangular display of complex numbers (internal value is RECT) (FF X (TAM) ; Info : SBI depends on SBcpx)	Setting (pgm)	ON
REGS	REGS	Register browser	Browse all registers (+: switch register/variable viewR/S: switch contents/storage viewRCL: recall bottom itemUp/Dn; A..D; I..L; 00..99: navigation)	Browser	
RESET	RESET	Reset	Reset the calculator (Startup using autosaved backup file C47auto.sav)	Command	
ResetF	ResetF	Reset fitting	Reset curve fitting	Command	
RE→CX	RE→CX	Real to complex	Convert reals to complex (in POLAR, using angle tag or ADM)	Function (dyadic)	
Re↔Im	Re↔Im	Exchange real and imaginary part	Exchange real and imaginary part	Command	
RJ	RJ	Right justify	Right justify (within word size) ; returns shift in X and result in Y	Function (monadic)	
RL	RL	Rotate left	Rotate left with number of bits (trailing input) (RL __ TamNonReg menu)	Function (monadic)	
RL1	RL1	Rotate left (1)	Shortcut to rotate left (1 bit)	Function (monadic)	
RLC	RLC	Rotate left through Carry	Rotate left through Carry (RLC __ TamNonReg menu)	Function (monadic)	
RMD	RMD	Remainder	Remainder of division of Y by X	Function (dyadic)	
RMODE ₀	RMODE	Rounding mode	Set floating point rounding mode, used only for RSD ; also used when converting from the extended precision internal format to packed reals (RMODE _ TamNonReg menu)	Setting (pgm)	0
RMODE?	RMODE?	Rounding mode	Floating point rounding mode ; set by RMODE	Command	
RNG _{6,145}	RNG	Range	Range 10 ⁿⁿ (min 99, max 6145) (RNG ____ TamNonReg menu)	Setting	6145
RNORM	RNORM	Row norm	Row norm	Command	
RootF []	RootF	Root fit	Root curve fitting	Setting	OFF
ROUND	ROUND	Round	Rounds to current display format (type real)	Function (monadic)	
ROUNDI	ROUNDI	Round to integer	Rounds to next integer (max 1000 digits)	Function (monadic)	
RPER. (+)	PER.	Radix period	Radix decimal period	Setting (pgm)	ON
RP _{HP} [·]	RP _{HP}	Classic Rect/Polar	Set for the classic (HP) stack conventions for →RECT and →POLAR ; Clear to follow C47 conventions of CC, COMPLEX and <i>i</i> (swapped) (Classic means X = x (Re) , Y = y (Im) for RECT ; X = r, Y = θ for POLAR)	Setting	ON
RPN	eRPNoff	Entry RPN off	Set stack to classic RPN mode (For programming purposes)	Setting (pgm)	

Catalog	Label	FullName	Description	Type	Default
RR	RR	Rotate right	Rotate right with number of bits (trailing input) (RR __ TamNonReg menu)	Function (monadic)	
RR1	RR1	Rotate right (1)	Shortcut to rotate right (1 bit)	Function (monadic)	
RRC	RRC	Rotate right through Carry	Rotate right through Carry (RRC __ TamNonReg menu)	Function (monadic)	
RSD	RSD	Round to significant digits	Rounds to number of significant digits, subject to rounding mode (RMODE) (RSD __ TamNonReg menu)	Function (monadic)	
RSUM	RSUM	Row sum	Row sum	Command	
RTN	RTN	Return	Return from (sub)routine to calling routine	Command (PEM)	
RTN+1	RTN+1	Return 2 levels up	Return from (sub)routine to 1 level higher than calling routine	Command (PEM)	
RWCOM, ()	WCOM,	Radix wide comma	Radix decimal wide comma	Setting (pgm)	OFF
RWDOT • ()	WDOT •	Radix wide dot	Radix decimal wide dot	Setting (pgm)	OFF
RWPER. ()	WPER.	Radix wide period	Radix decimal wide period	Setting (pgm)	OFF
R-CLR	R-CLR	Clear registers	Clear registers (sss.nn means clear registers from sss through sss + nn - 1)	Command	
R-COPY	R-COPY	Copy registers	Copy registers (sss.nnnnn means copy registers from sss through sss + nn - 1 to registers ddd through ddd + nn - 1)	Command	
R-SORT	R-SORT	Sort registers	Sort registers (sss.nn means sort registers from sss through sss + nn - 1)	Command	
R-SWAP	R-SWAP	Swap registers	Swap registers (sss.nnnnn means swap registers from sss through sss + nn - 1 with registers ddd through ddd + nn - 1)	Command	
R↑	R↑	Roll up	Roll up stack	Command	
R↓	R↓	Roll down	Roll down stack	Command	
s	s	Sample standard deviation	Sample standard deviation (s_x ; s_y = (2 stack levels))	Command	
SAVE	SAVE	Save full backup	Save full backup to file C47.sav in FAT (SAVFILES ; Inverse : LOAD)	Command (nonpgm)	
SAVEST	SAVEST	Save state file	Save state file to file <state.s47> in FAT (File save dialog (STATE) ; Inverse : LOADST)	Command (nonpgm)	
SB	SB	Set bit	Set bit n (SB __ TamNonReg menu)	Function (monadic)	
SCATR	SCATR	Scatter plot	Scatter plot of measurements	MENU (item)	
SCI ()	SCI	Scientific notation	Set numeric display mode to Scientific notation with nn+1 digits (SCI __ TamNonReg menu)	Setting (pgm)	OFF
SCIOVR (•)	SCIOVR	Scientific display large reals	Change display to SCI for reals too large to display in full (FF A)	Setting (pgm)	ON
SDIGS ₃₄	SDIGS	Set significant digits	Set the number of significant digits (1 ... 34) for rounding after each operation ; sets tolerance of Solver and CONVG? ; value of 0 sets maximum precision (34) (SDIGS __ TamNonReg menu)	Setting (pgm)	34
SDIGS?	SDIGS?	Significant Digits	Number of significant digits ; set by SDIGS	Command	
SDL	SDL	Shift Digits Left	Shift digits to the left (SDL __ TamNonReg menu)	Function (monadic)	
SDR	SDR	Shift Digits Right	Shift digits to the right (SDR __ TamNonReg menu)	Function (monadic)	
SEC	SEC	Seconds	Seconds (of time)	Function (monadic)	
SEED	SEED	Seed	Set random seed (0..1] ; for values less than or equal to 0, the seed is derived from the internal clock	Command	
SETCHN	CHINA	Chinese formatting	Set to Chinese regional formats (date, time, calendar, number formatting) (First Gregorian day set: 1949-10-01)	Setting (pgm)	
SETDAT	SETDAT	Set date	Set date	Setting (pgm)	
SETDFLT	DFLT	Default regional formatting	Set to default regional formats (date, time, calendar, number formatting) (First Gregorian day set: 1752-09-14)	Setting (pgm)	
SETEUR	EUROPE	European formatting	Set to European regional formats (date, time, calendar, number formatting) (First Gregorian day set: 15.10.1582)	Setting (pgm)	
SETIND	INDIA	Indian formatting	Set to Indian regional formats (date, time, calendar, number formatting) (First Gregorian day set: 14.09.1752)	Setting (pgm)	
SETJPN	JAPAN	Japanese formatting	Set to Japanese regional formats (date, time, calendar, number formatting) (First Gregorian day set: 1873-01-01)	Setting (pgm)	
SETTIM	SETTIM	Set time	Set time	Setting (pgm)	
SETUK	UK	UK formatting	Set to UK regional formats (date, time, calendar, number formatting) (First Gregorian day set: 14.09.1752)	Setting (pgm)	
SETUSA	USA	USA formatting	Set to USA regional formats (date, time, calendar, number formatting) (First Gregorian day set: 9/14/1752)	Setting (pgm)	
set→TXT	set→TXT	Settings to text file	Settings to text file	Command (strike)	
SF	SF	Set flag	Set flag (SF __ TamFlag menu)	Command	
SHOW	SHOW	Show	Show item in maximum detail, favouring register data type (tag)	Command	
SH.4s [•]	SH.4s	Shift time-out 4s	Set shift to time out after 4 seconds	Setting	ON
SIG ()	SIG	Significant digits notation	Set numeric display mode to Significant notation with nn+1 digits ; switching over to scientific or engineering notation when number of rounded trailing zeros exceeds group size (IPGRP) (SIG __ TamNonReg menu)	Setting (pgm)	OFF

Catalog	Label	FullName	Description	Type	Default
sign	sign	Sign	Sign is -1 for negative numbers, 0 for zero, +1 for positive numbers	Function (monadic)	
SIGNMT ()	SIGNMT	Sign and mantissa	Set sign and mantissa mode for shortint (SBI depends on SBint)	Setting (pgm)	OFF
SIM_EQ	SIM EQ	Matrix simultaneous equations	Matrix simultaneous equations functions (Mat A * Mat X = Mat B) (SIM_EQ __ TamNonReg menu ; CAT.MENUS M.SIMQ)	MENU (item)	
SIN	SIN	Sine	Sine	Function (monadic)	
sinc	sinc	Sinc	(Sine of X) / X	Function (monadic)	
sincπ	sincπ	Sinc pi	(Sine of π * X) / (π * X)	Function (monadic)	
sinh	sinh	Hyperbolic sine	Hyperbolic sine	Function (monadic)	
SKIP	SKIP	Skip	Skip n program steps (SKIP __ TamNonReg menu)	Command (PEM)	
SL	SL	Shift left	Shift bits right (SL __ TamNonReg menu)	Function (monadic)	
SL1	SL1	Shift left (1)	Shortcut to shift left (1 bit)	Function (monadic)	
SLVQ	SLVQ	SLVQ	Solves the quadratic equation (parameters X = a, Y = b, Z = c)	Command	
s _m	s _m	Standard error of the mean	Standard error of the mean (s _{mX} ; s _{mY} = (2 stack levels))	Command	
s _{mi}	s _{mi}	Precision	Precision of measuring instrument investigated, requires 30 data pairs (s _{mi} =)	Command	
s _{mW}	s _{mW}	Standard error of the weighted mean	Standard error of the weighted mean (s _{mW} =)	Command	
SNAP	SNAP	Screenshot	Save screenshot as image to bitmap file in FAT ; if executed from the NORMAL keyboard (C47.81.13) saves contents of stack or alpha buffer as text to data file in FLASH memory ; plays clicking sound ([F/G] + [E])	Command	
sn(u,m)	sn(u,m)	Elliptic sine	Elliptic sine (u = X ; m = Y)	Function (dyadic)	
SOLVE	SOLVE	Solve	Solve the equation ; use X, Y as initial guesses ; fill all stack registers with X (SOLVE __ TamLbl(Alpha) menu)	Command	
SPCRES [+]	SPCRES	Special results	Set to allow special results of calculations (infinity, not-a-number) ; an error will not occur for such events (FF D)	Setting	ON
SPCRES0	SPCRES0	Normal results	Do not allow special results of calculations (infinity, not-a-number) ; an error will occur for such events (For programming purposes)	Command	
SPCRES1	SPCRES1	Special results	Allow special results of calculations (infinity, not-a-number) ; an error will not occur for such events (For programming purposes)	Command	
SPEC?	SPEC?	Special?	Test X is special (∞ or NaN)	Function (monadic)	
Sqr	SQR	Square root	Square root	Function (legacy, monadic)	
SR	SR	Shift right	Shift bits right (SR __ TamNonReg menu)	Function (monadic)	
SR1	SR1	Shift right (1)	Shortcut to shift right (1 bit)	Function (monadic)	
SSIZE4 ()	SSIZE4	Stack size 4	Set stack size to 4 registers (SBI depends on SBss)	Setting (pgm)	OFF
SSIZE8 (+)	SSIZE8	Stack Size 8	Set stack size to 8 registers (SBI depends on SBss)	Setting (pgm)	ON
SSIZE?	SSIZE?	Stack size	Number of stack registers currently allocated (4 or 8) ; set by SSIZE4 ; SSIZE8	Command	
STATUS	STATUS	Status	Show status pages (memory, flags and settings) (Flags)	Command	
ST0	ST0	Store (register)	Store value in register or variable ; can be followed by +, -, ×, ÷ for add into, subtract into, multiply into, divide into functions (ST0 __ TamStoRcl(Alpha) menu)	Command	
ST0 3I	ST0 3I	Store triple I	Copy X, Y, Z to R96, R97, R98	Command	
ST0 3V	ST0 3V	Store triple V	Copy X, Y, Z to R93, R94, R95	Command	
ST0 3Z	ST0 3Z	Store triple Z	Copy X, Y, Z to R90, R91, R92	Command	
STOCFG	STOCFG	Store configuration	Store configuration in register or variable (STOCFG __ TamStoRcl(Alpha) menu)	Command	
STOEL	STOEL	Store X into current element	Store X into current element	Command	
ST0IJ	ST0IJ	Set current index	Set current index	Command	
STOP	STOP	Run/Stop	Run/Stop (Program)	Command	
STOPW	STOPW	Stopwatch	Stopwatch	App (item)	
STOS	STOS	Store stack	Store entire stack in 4 or 8 registers (STOS __ TamStoRcl(Alpha) menu)	Command	
ST0+	ST0+	Add into	Add X to register or variable (ST0+ __ TamStoRcl(Alpha) menu)	Function (monadic)	
ST0-	ST0-	Subtract into	Subtract X from register or variable (ST0- __ TamStoRcl(Alpha) menu)	Function (monadic)	
ST0×	ST0×	Multiply into	Multiply register or variable by X (ST0× __ TamStoRcl(Alpha) menu)	Function (monadic)	
ST0/	ST0/	Divide into	Divide register or variable by X (ST0/ __ TamStoRcl(Alpha) menu)	Function (monadic)	
ST0↑	ST0↑	Store maximum	Store maximum of X and register or variable (ST0↑ __ TamStoRcl(Alpha) menu)	Command	
ST0↓	ST0↓	Store minimum	Store minimum of X and register or variable (ST0↓ __ TamStoRcl(Alpha) menu)	Command	
STRI?	STRI?	String?	Test X is text string	Function (monadic)	
s _w	s _w	Weighted population standard deviation	Weighted population standard deviation (s _w =)	Command	
s _{xy}	s _{xy}	Sample covariance	Sample covariance	Command	

Catalog	Label	FullName	Description	Type	Default
SYMtoA	→ abc	Convert from symmetrical components	Convert symmetrical components a0, a1, a2 in Z, Y, X to 3-phase a, b, c in Z, Y, X (→ 012)	Function (triadic)	
SYSTEM	SYSTEM	System (exit)	Exit calculator (causing reset) and enter DMCP (hardware only) ; confirmation dialog "Are you sure?" [Y/N] (Autosaves backup file C47auto.sav to FAT)	Command	
s(a)	s(a)	Standard errors	Standard errors of line fitted ; s(a ₀) in X ; s(a ₁) in Y (Works for EXPF, LINF, LOGF, ORTOHOF, POWERF ; does not work for CAUCHF, GAUSSF, HYPF, PARABF, ROOTF)	Command	
S.RESET	S.RESET	Reset f/g timers	Safe reset, then toggle ON/OFF all accessibility related options: HOME.3 [*] ; g.2Tp [*] ; SH.4s [*] ; fg.FUL [*]	Command	
TAN	TAN	Tangent	Tangent	Function (monadic)	
tanh	tanh	Hyperbolic tangent	Hyperbolic tangent	Function (monadic)	
TDISP ₀	TDISP	Time display format	Set time display format: 0 = full ; 1,2 = hours/minutes ; 3 = hours/minutes/seconds ; 4,5,6 = 1,2,3 decimal places for the seconds (TDISP _ TamNonReg menu)	Setting	0
TICKS	TICKS	Ticks	Number of ticks counted since calculator was turned on (one tick is 10 ms)	Command	
TIME	TIME	Time	Current time	Command	
TIME→	TIME→	Time to stack	Convert time to hours (24h), minutes, seconds in stack	Function (monadic)	
T _n	T _n	Chebyshev polynomials (1st)	Chebyshev polynomials of the 1st kind (x = X ; n = Y)	Function (dyadic)	
TOPE	TOPE	Tone	Tone (0-11) (TONE _ TamNonReg menu)	Command	
TOP?	TOP?	Top?	Test program pointer is in top routine (as opposed to in subroutine)	Command	
t _p (x)	t _p (x)	Student's t pdf	Student's t probability density function	Function (tbd)	
TRANS	TRANS	Transpose matrix	Transpose matrix	Function (legacy)	
t _l (x)	t _l (x)	Student's t cdf (lower)	Student's t cumulative distribution (lower tail)	Function (tbd)	
t _u (x)	t _u (x)	Student's t cdf (upper)	Student's t cumulative distribution (upper tail)	Function (tbd)	
t ⁻¹ (p)	t ⁻¹ (p)	Student's t (inverse)	Student's t probability inverse function	Function (tbd)	
t _z	t _z	Swap T	Swap T and register (t _z _ Tam menu)	Command	
ULP?	ULP?	Unit in the last place	Minimum difference to next or previous machine representable real, as power of ten	Command	
U _n	U _n	Chebyshev polynomials (2nd)	Chebyshev polynomials of the 2nd kind (x = X ; n = Y)	Function (dyadic)	
UNDO	↶	Undo	Restore complete stack and LASTx register	Command	
UNIT ()	UNIT	Unit (prefix) notation	Set numeric display mode to UNIT nn+1 digits ; setting for showing all or limited set of prefixes: PFX.All (display using prefix also available from HOME for numeric entry: p - n - μ - m - k - M - G - T for pico, nano, micro, milli, kilo, Giga, Tera) (UNIT _ TamNonReg menu)	Setting (pgm)	OFF
UNITV	UNITV	Unit vector	Unit vector for complex number or matrix	Command	
UNSIGN ()	UNSIGN	Unsigned	Set unsigned mode for shortint (SBI depends on SBint)	Setting (pgm)	OFF
VARMNU	VarMNU	Variable menu	Create variable menu (VARMNU _ TamLbl(Alpha) menu)	MENU (item ; PEM)	
VERS?	VERS?	Version	Show firmware version (Firmware version)	Command (nonpgm)	
VIEW	VIEW	View	View register or variable (VIEW _ Tam menu)	Command	
VOL ₁₁	VOL	Volume	Set audio volume (0-11) (VOL _ TamNonReg menu)	Setting (pgm)	11
VOL?	VOL?	Volume?	Audio volume (0-11)	Command	
VOL↑ ₁₁	VOL↑	Volume up	Volume up (0-11)	Setting	11
VOL↓ ₁₁	VOL↓	Volume down	Volume down (0-11)	Setting	11
V _z	V _z	Vector angle	Angle between two vectors (2D or 3D)	Function (dyadic)	
WDAY	WDAY	Weekday	Show weekday for date (Weekday)	Function (monadic)	
Weibl _p	Weibl _p	Weibull pdf	Weibull probability density function	Function (tbd)	
Weibl _l	Weibl _l	Weibull cdf (lower)	Weibull cumulative distribution (lower tail)	Function (tbd)	
Weibl _u	Weibl _u	Weibull cdf (upper)	Weibull cumulative distribution (upper tail)	Function (tbd)	
Weibl ⁻¹	Weibl ⁻¹	Weibull (inverse)	Weibull probability inverse function	Function (tbd)	
WHO?	WHO?	Who	Show calculator development team names (Team names)	Command (nonpgm)	
W _m	W _m	W _m	Lambert's W function (negative branch ; m = minus)	Function (tbd)	
W _p	W _p	W _p	Lambert's W function (principal branch)	Function (tbd)	
WRITEP	WRITEP	Write program	Write program to file <program.p47> in FAT (WRITEP _ TamLabel ; DMCP : File save dialog (PROGRAMS) ; Inverse : READP)	Command	
WSIZE ₆₄	WSIZE	Word size	Set word size for shortint (WSIZE _ TamNonReg menu ; Info : SBI depends on SBint)	Setting (pgm)	64
WSIZE?	WSIZE?	Word size?	Word size for short integers ; set by WSIZE	Command	
W ⁻¹	W ⁻¹	W ⁻¹	Inverse of W _p (≥ -1)	Function (tbd)	
W.SWP	W.SWP	Word swap	Swap words	Function (monadic)	
x̂	x̂	x hat	Estimation of x	Command	
x̄	x̄	Arithmetic means	Arithmetic means (x̄ ; ȳ = (2 stack levels))	Command	

Catalog	Label	FullName	Description	Type	Default
X → BAL	X → BAL	X Balanced	Create balanced 3 phase quantities by pushing onto stack X * a, and then X * a * a	Function (monadic)	
x ²	x ²	Square	Square of X	Function (monadic)	
x ³	x ³	Cube	Raise to power of 3	Function (monadic)	
XEQ	XEQ	Execute	Execute function or program (XEQ __ TamLb(Alpha) menu)	Command	
XEQM01	XEQM01	XEQ XEQM01	Execute XEQM01 (HELP!!)	Command	
XEQM02	XEQM02	XEQ XEQM02	Execute XEQM02 (BATPLT)	Command	
XEQM03	XEQM03	XEQ XEQM03	Execute XEQM03 (MP2203)	Command	
XEQM04	XEQM04	XEQ XEQM04	Execute XEQM04 (MP2281)	Command	
XEQM05	XEQM05	XEQ XEQM05	Execute XEQM05 (MP3217)	Command	
XEQM06	XEQM06	XEQ XEQM06	Execute XEQM06 (CUBES)	Command	
XEQM07	XEQM07	XEQ XEQM07	Execute XEQM07 (GUDERM)	Command	
XEQM08	XEQM08	XEQ XEQM08	Execute XEQM08 (PYTHAG)	Command	
XEQM09	XEQM09	XEQ XEQM09	Execute XEQM09 (PLTPRIM)	Command	
XEQM10	XEQM10	XEQ XEQM10	Execute XEQM10 (58TESTS)	Command	
XEQM11	XEQM11	XEQ XEQM11	Execute XEQM11 (SINC_PI)	Command	
XEQM12	XEQM12	XEQ XEQM12	Execute XEQM12 (BINET)	Command	
XEQM13	XEQM13	XEQ XEQM13	Execute XEQM13 (TRAPZ)	Command	
XEQM14	XEQM14	XEQ XEQM14	Execute XEQM14 (PLTFOR)	Command	
XEQM15	XEQM15	XEQ XEQM15	Execute XEQM15 (X15)	Command	
XEQM16	XEQM16	XEQ XEQM16	Execute XEQM16 (BINETF)	Command	
XEQM17	XEQM17	XEQ XEQM17	Execute XEQM17 (RANDOM)	Command	
XEQM18	XEQM18	XEQ XEQM18	Execute XEQM18 (TEST)	Command	
\bar{x}_G	\bar{x}_G	Geometric means	Geometric means (\bar{x}_G ; $\bar{y}_G = (2 \text{ stack levels})$)	Command	
\bar{x}_H	\bar{x}_H	Harmonic means	Harmonic means (\bar{x}_H ; $\bar{y}_H = (2 \text{ stack levels})$)	Command	
x _{IQR}	x _{IQR}	Interquartile range	Interquartile range for both X and Y ; this is equal to Q.3 - Q.1 (iqr _x ; iqr _y = (2 stack levels))	Command	
x _{MAD}	x _{MAD}	Median absolute deviation	Median absolute deviation for both X and Y ; this is the median of the differences between each data point and the overall median (mad _x ; mad _y = (2 stack levels))	Command	
x _{MAX}	x _{MAX}	Maxima	Maxima (x _{MAX} ; y _{MAX} = (2 stack levels))	Command	
x _{MEDN}	x _{MEDN}	Median	Sort the data and return the middle value for both X and Y ; for an even number of samples, the arithmetic mean of the two middle values is returned (md _x ; md _y = (2 stack levels))	Command	
x _{MIN}	x _{MIN}	Minima	Minima (x _{MIN} ; y _{MIN} = (2 stack levels))	Command	
XNOR	XNOR	Exclusive NOR	Logical exclusive NOR (bitwise)	Function (dyadic)	
XOR	XOR	XOR	Logical exclusive OR	Function (dyadic)	
XPORTP	XPORTP	Export program	Export program to text file in FAT (XPORTP _ TamLabel ; DMCP : File save dialog (PROGRAMS))	Command	
x _{Q1}	x _{Q1}	Lower quantile	Lower quantile for both X and Y (Q _{1 x} ; Q _{1 y} = (2 stack levels))	Command	
x _{Q3}	x _{Q3}	Upper quantile	Upper quantile for both X and Y (Q _{3 x} ; Q _{3 y} = (2 stack levels))	Command	
x _{RANGE}	x _{RANGE}	Stats range	Range for both X and Y ; this is equal to MAX - MIN of the statistics matrix (STATS) (rg _x ; rg _y = (2 stack levels))	Command	
\bar{x}_{RMS}	\bar{x}_{RMS}	Quadratic means	Quadratic means (root mean square) (\bar{x}_{RMS} ; $\bar{x}_{RMS} = (2 \text{ stack levels})$)	Command	
x _{SUM}	x _{SUM}	x _{SUM}	Return Σx and Σy in X and Y respectively (Σ_x ; $\Sigma_y = (2 \text{ stack levels})$)	Command	
\bar{x}_w	\bar{x}_w	Weighted mean	Weighted means of x with weight y ($\bar{x}_w =$)	Command	
$\sqrt[x]{y}$	$\sqrt[x]{y}$	xth root	Xth root of Y	Function (dyadic)	
X.SWAP	X.SWAP	Swap X	Swap contents of X register with input of alpha or equation editor	Command	
X.XEQ	X.XEQ	Execute XEQM command	Execute XEQM command in X-register	Command	
x!	x!	Factorial ; $\Gamma(x+1)$	For integers : x! ; for reals : $\Gamma(x+1)$ (Max integer: 450 ; max real : 2123.549 956 662 463 236 31 ; integers > max are converted to reals)	Function (monadic)	

Catalog	Label	FullName	Description	Type	Default
x→DATE	x→DATE	X to date	Convert date input number YYYY-MM-DD or DD.MM.YYYY or MM/DD/YYYY to date according to DISP or CLK format settings (Weekday)	Function (monadic)	
x→α	x→α	X to alpha	Convert character code in X to alpha character (code value pushed to Y)	Function (monadic)	
x↔	x↔	Swap X	Swap X and register (x↔ __ Tam menu)	Command	
x↔y	x↔y	Swap X and Y	Swap register X and register Y	Command	
x< ?	x< ?	X less?	X less? (x< ? __ TamCmp menu)	Function (monadic)	
x≤ ?	x≤ ?	X less or equal?	X less or equal? (x≤ ? __ TamCmp menu)	Function (monadic)	
x= ?	x= ?	X equals?	X equals? (x= ? __ TamCmp menu)	Function (monadic)	
x≈ ?	x≈ ?	X approximates?	X approximates? (rounded values are equal) (x≈ ? __ TamCmp menu)	Function (monadic)	
x≠ ?	x≠ ?	X not equal?	X not equal? (x≠ ? __ TamCmp menu)	Function (monadic)	
x≥ ?	x≥ ?	X greater or equal?	X greater or equal? (x≥ ? __ TamCmp menu)	Function (monadic)	
x> ?	x> ?	X greater?	X greater? (x> ? __ TamCmp menu)	Function (monadic)	
x=+0 ?	x=+0 ?	X equals +0?	X equals +0? (shortint ICOMPL or SIGNMT) (x=+0 ? __ TamCmp menu)	Function (monadic)	
x=-0 ?	x=-0 ?	X equals -0?	X equals -0? (shortint ICOMPL or SIGNMT) (x=-0 ? __ TamCmp menu)	Function (monadic)	
x%ILE	x%ILE	Percentile	Percentile (pctile _x ; pctile _y = (2 stack levels))	Command	
ŷ	ŷ	y hat	Estimation of y	Command	
Y → Δ	Y → Δ	Star (Wye) to Delta	Convert delta connected impedances X, Y, Z to star impedances X, Y, Z (Δ → Y)	Function (triadic)	
YEAR	YEAR	Year	Year (of date)	Command	
YMD (•)	YMD	Year month day	Date display mode YYYY-MM-DD (YYYY-MM-DD)	Setting (pgm)	ON
y ^x	y ^x	y to the power x	Raise value in the Y-register to the power in the X-register	Function (dyadic)	
Y _y (x)	Y _y (x)	Y _y (x)	Bessel function of the 2nd kind and order y	Command	
y↔	y↔	Swap Y	Swap Y and register (y↔ __ Tam menu)	Command	
zyx→M	zyx→M	Compose 3x1 matrix	Create 3x1 matrix from ZYX (M→zyx)	Function (triadic)	
z↔	z↔	Swap Z	Swap Z and register (z↔ __ Tam menu)	Command	
αLENG?	αLENG?	Alpha length	String length (αLENG? __ Tam menu)	Function (monadic)	
αPOS?	αPOS?	Alpha position	Position in string in variable or stack of substring in X (starting from 0) (αPOS? __ Tam menu)	Function (monadic)	
αRL	αRL	Alpha rotate left	Alpha rotate left (αRL __ Tam menu)	Function (monadic)	
αRR	αRR	Alpha rotate right	Alpha rotate right (αRR __ Tam menu)	Function (monadic)	
αSL	αSL	Alpha shift left	Alpha shift left (αSL __ Tam menu)	Function (monadic)	
αSR	αSR	Alpha shift right	Alpha shift right (αSR __ Tam menu)	Function (monadic)	
αPARSE	αPARSE	Alpha parse	Parse alpha input for numeric content (in development)	Command (strike)	
α→x	α→x	Alpha to X	Convert first alpha character of string in variable or register to character code (hexadecimal) (α→x __ Tam menu)	Function (monadic)	
β(x,y)	β(x,y)	Euler's Beta function	Euler's Beta function	Function (dyadic)	
Γ _{xy}	Γ _{xy}	Γ _{xy}	Upper incomplete Gamma function	Function (dyadic)	
γ _{xy}	γ _{xy}	γ _{xy}	Lower incomplete Gamma function	Function (dyadic)	
Γ(x)	Γ(x)	Γ function	Gamma function	Function (monadic)	
Δ → Y	Δ → Y	Delta to Star (Wye)	Convert star connected impedances X, Y, Z to delta impedances X, Y, Z (Y → Δ)	Function (triadic)	
Δ%	Δ%	Delta percent	Delta percentage from Y to X, keeping Y on stack (Δ% :)	Function (dyadic)	
Δ% \bar{x}	Δ% \bar{x}	Delta percentage to mean	Delta percentage from \bar{x} to x using statistics matrix (STATS) (Δ% :)	Function (monadic)	
ε	ε	Scattering factor for a lognormal sample	Scattering factor for a lognormal sample (ε _x ; ε _y = (2 stack levels))	Command	
ε _m	ε _m	Scattering factor of the geometric mean	Scattering factor of the geometric mean (ε _{m_x} ; ε _{m_y} = (2 stack levels))	Command	
ε _p	ε _p	Scattering factor for a lognormal population	Scattering factor for a lognormal population (ε _{p_x} ; ε _{p_y} = (2 stack levels))	Command	
ζ(x)	ζ(x)	ζ(x)	Riemann's Zeta for real arguments	Function (monadic)	
ζ(φ,m)	ζ(φ,m)	Jacobi's Zeta	Jacobi's Zeta (φ = X ; m = Y)	Function (dyadic)	
Π _n	Π _n	Product (programmable)	Real or complex product using specified program, with iteration counter, interrupt by keypress (Π _n __ TamLb(Alpha) menu ; <from> ENTER <to> ENTER <step>)	Command	
Π(n,m)	Π(n,m)	Complete elliptic integral (2rd)	Complete elliptic integral of the 3rd kind (n = X ; m = Y)	Function (dyadic)	
σ	σ	Population standard deviation	Population standard deviation (σ _x ; σ _y = (2 stack levels))	Command	
Σln ² x	Σln ² x	Summation of ln ² x	Summation of ln ² x using statistics matrix (STATS)	Command	
Σln ² y	Σln ² y	Summation of ln ² y	Summation of ln ² y using statistics matrix (STATS)	Command	

Catalog	Label	FullName	Description	Type	Default
$\Sigma \ln x$	$\Sigma \ln x$	Summation of $\ln x$	Summation of $\ln x$ using statistics matrix (STATS)	Command	
$\Sigma \ln x \cdot \ln y$	$\Sigma \ln x \cdot \ln y$	Summation of $\ln x \cdot \ln y$	Summation of $\ln x \cdot \ln y$ using statistics matrix (STATS)	Command	
$\Sigma \ln y$	$\Sigma \ln y$	Summation of $\ln y$	Summation of $\ln y$ using statistics matrix (STATS)	Command	
Σ_n	Σ_n	Sum (programmable)	Real or complex sum using specified program, with iteration counter, interrupt by keypress (Σ_n __)	Command	
σ_w	σ_w	Weighted population standard deviation	Weighted population standard deviation ($\sigma_w =$)	Command	
Σx	Σx	Summation of x	Summation of x using statistics matrix (STATS)	Command	
Σx^2	Σx^2	Summation of x^2	Summation of x^2 using statistics matrix (STATS)	Command	
$\Sigma x^2 y$	$\Sigma x^2 y$	Summation of $x^2 y$	Summation of $x^2 y$ using statistics matrix (STATS)	Command	
$\Sigma x^2 y^{-1}$	$\Sigma x^2 y^{-1}$	Summation of $x^2 y^{-1}$	Summation of $x^2 y^{-1}$ using statistics matrix (STATS)	Command	
$\Sigma x^2 \cdot \ln y$	$\Sigma x^2 \cdot \ln y$	Summation of $x^2 \cdot \ln y$	Summation of $x^2 \cdot \ln y$ using statistics matrix (STATS)	Command	
Σx^3	Σx^3	Summation of x^3	Summation of x^3 using statistics matrix (STATS)	Command	
Σx^4	Σx^4	Summation of x^4	Summation of x^4 using statistics matrix (STATS)	Command	
Σxy	Σxy	Summation of xy	Summation of xy using statistics matrix (STATS)	Command	
Σxy^{-1}	Σxy^{-1}	Summation of xy^{-1}	Summation of xy^{-1} using statistics matrix (STATS)	Command	
Σx^{-1}	Σx^{-1}	Summation of x^{-1}	Summation of x^{-1} using statistics matrix (STATS)	Command	
$\Sigma x^{-1} \cdot \ln y$	$\Sigma x^{-1} \cdot \ln y$	Summation of $x^{-1} \cdot \ln y$	Summation of $x^{-1} \cdot \ln y$ using statistics matrix (STATS)	Command	
Σx^{-2}	Σx^{-2}	Summation of x^{-2}	Summation of x^{-2} using statistics matrix (STATS)	Command	
$\Sigma x \cdot \ln y$	$\Sigma x \cdot \ln y$	Summation of $x \cdot \ln y$	Summation of $x \cdot \ln y$ using statistics matrix (STATS)	Command	
Σy	Σy	Summation of y	Summation of y using statistics matrix (STATS)	Command	
Σy^2	Σy^2	Summation of y^2	Summation of y^2 using statistics matrix (STATS)	Command	
Σy^{-1}	Σy^{-1}	Summation of y^{-1}	Summation of y^{-1} using statistics matrix (STATS)	Command	
Σy^{-2}	Σy^{-2}	Summation of y^{-2}	Summation of y^{-2} using statistics matrix (STATS)	Command	
$\Sigma y \cdot \ln x$	$\Sigma y \cdot \ln x$	Summation of $y \cdot \ln x$	Summation of $y \cdot \ln x$ using statistics matrix (STATS)	Command	
$\Sigma +$	$\Sigma +$	Sigma+	Enter data into the statistics matrix (STATS) (nnn data points)	Command	
$\Sigma -$	$\Sigma -$	Sigma-	Remove data from the statistics matrix (STATS) (nnn data points)	Command	
ϕ_p	ϕ_p	Standard normal pdf	Standard probability density function	Function (tbd)	
ϕ_{\downarrow}	ϕ_{\downarrow}	Standard normal cdf (lower)	Standard cumulative distribution (lower tail)	Function (tbd)	
ϕ_{\uparrow}	ϕ_{\uparrow}	Standard normal cdf (upper)	Standard cumulative distribution (upper tail)	Function (tbd)	
ϕ^{-1}	ϕ^{-1}	Standard normal (inverse)	Standard probability inverse function	Function (tbd)	
$\chi^2_p(x)$	$\chi^2_p(x)$	$\chi^2_p(x)$	χ^2 probability density function	Function (tbd)	
$\chi^2_{\downarrow}(x)$	$\chi^2_{\downarrow}(x)$	$\chi^2_{\downarrow}(x)$	χ^2 cumulative distribution (lower tail)	Function (tbd)	
$\chi^2_{\uparrow}(x)$	$\chi^2_{\uparrow}(x)$	$\chi^2_{\uparrow}(x)$	χ^2 cumulative distribution (upper tail)	Function (tbd)	
$\psi(u,m)$	$\psi(u,m)$	Elliptic amplitude	Elliptic amplitude ($u = X$; $m = Y$)	Function (dyadic)	
$(\chi^2)^{-1}$	$(\chi^2)^{-1}$	$(\chi^2)^{-1}$	χ^2 probability inverse function	Function (tbd)	
$(-1)^x$	$(-1)^x$	$(-1)^x$	Cosine of πx [+ i * sine of πx for complex parameter] (Esp. for non-integer x)	Function (monadic)	
$[M]^T$	$[M]^T$	Transpose matrix	Transpose matrix	Function (monadic)	
$[M]^{-1}$	$[M]^{-1}$	Invert matrix	Inverse of matrix	Function (monadic)	
$+$	$[+]$	Add	Add X to Y ; concatenate X and Y (Concatenation includes numbers, dates and times + strings and vice versa)	Function (dyadic)	
$\pm\omega?$	$\pm\omega?$	Infinities	Show whether X -register contains positive or negative infinite value	Command	
$-$	$[-]$	Subtract	Subtract X from Y	Function (dyadic)	
\times	$[\times]$	Multiply	Multiply Y by X (215)	Function (dyadic)	
$\times\text{MOD}$	$\times\text{MOD}$	$Z \times Y$ modulo X	$Z \times Y$, modulo X	Function (triadic)	
$\wedge\text{MOD}$	$\wedge\text{MOD}$	Z^Y modulo X	Z^Y modulo X	Function (triadic)	
$.ms$	$.ms$	Minutes & seconds	Convert sexagesimal format input sequence or decimal stack value to hh:mm:ss hours or dd°mm'ss" degrees (cyclic) (NIM input treated as sexagesimal (hh/dd.mmss) format; stack input treated as decimal value)	Function (cyclic; monadic)	
$.ms^{-1}$	$.ms^{-1}$	Inverse of $.ms$	Convert hh:mm:ss hours or dd°mm'ss" degrees to sexagesimal format number (untag) (hh/ddd.mmss:)	Function (monadic)	
\div	$[\div]$	Divide	Divide Y by X (Obelus: 247; Solidus: 47)	Function (dyadic)	
$_HOME$ []	HOME	HOME menu shown	HOME menu shown (when all menus are exited using EXIT)	Setting	OFF
$\rightarrow\text{DATE}$	$\rightarrow\text{DATE}$	Stack to date	Convert stack values X , Y , Z to date according to DISP or CLK settings for date format (weekday)	Function (triadic)	
$\rightarrow\text{DEG}$	$\rightarrow\text{DEG}$	Set DEG tag or convert to DEG	If untagged, set tag to DEG; if tagged, convert X to degrees; does not change ADM (°)	Function (monadic)	
$\rightarrow\text{D.MS}$	$\rightarrow\text{D.MS}$	Set D.MS tag or convert to D.MS	If untagged, set tag to D.MS; if tagged, convert X to D.MS; does not change ADM; X considered as dd.mmss (° ' ")	Function (monadic)	
$\rightarrow\text{GRAD}$	$\rightarrow\text{GRAD}$	Set GRAD tag or convert to GRAD	If untagged, set tag to GRAD; if tagged, convert X to GRAD; does not change ADM (°)	Function (monadic)	
$\rightarrow\text{HR}$	$\rightarrow\text{HR}$	Convert to decimal hours	Convert to decimal hours (Assumes angle conforms to ADM)	Function (legacy; monadic)	

Catalog	Label	FullName	Description	Type	Default
→h.ms	→h.ms	Convert to hours, min, sec	Convert sexagesimal format input sequence or decimal stack value to hh:mm:ss hours (NIM input treated as sexagesimal (hh:mm:ss) format ; stack input treated as decimal value)	Function (monadic)	
→INT	→INT	Number (base)	Set number base ; operates on all stack registers depending on BASE _{HP} ; reset by [.d] (gShifted [LOG]) (Indirection (→) activates TamNonRegInd ; Shortcuts : H:16 (hex) ; ENTER or D:10 (shortint) ; O:8 (octal) ; B:2 (binary) ; Info : SBI depends on SBfrac)	Setting (pgm ; stack)	
→MULπ	→MULπ	Set MULπ tag or convert to MULπ	If untagged, set tag to MULπ ; if tagged, convert X to MULπ ; does not change ADM (π)	Function (monadic)	
→POLAR	→POLAR	To polar	Transform rectangular to polar coordinates (stack conventions according to flag RP _{HP} ; transform complex number to polar notation (monadic) and set POLAR tag (r = ; θ = (2 stack levels))	Function (monadic ; dyadic)	
→RAD	→RAD	Set RAD tag or convert to RAD	If untagged, set tag to RAD ; if tagged, convert X to RAD ; does not change ADM (r)	Function (monadic)	
→REAL	→REAL	Convert to real number	Convert to real number	Function (legacy)	
→RECT	→RECT	To rectangular	Transform polar to rectangular coordinates (stack conventions according to flag RP _{HP} ; transform complex number to rectangular notation (monadic) and set RECT tag (x : Re = ; y : Im = (2 stack levels))	Function (monadic ; dyadic)	
→TIME	→TIME	Stack to Time	Convert to time	Function (monadic)	
↑BIN	↑BIN	High bin	High bin (nBINS : ; ↑BIN : ; ↑BIN : (3 stack levels))	Command	
↻	↻	Shuffle stack	Shuffle stack; replace X Y Z T by selection of X Y Z T (↻ ____ TamShuffle menu)	Command	
↓BIN	↓BIN	Low bin	Low bin (nBINS : ; ↓BIN : ; ↑BIN : (3 stack levels))	Command	
		Parallel	Parallel impedance = (X × Y) / (X + Y)	Function (dyadic)	
M	M	Determinant	Determinant	Function (monadic)	
x	x	Magnitude	Magnitude (absolute value) of complex number	Function (monadic)	
%	%	Percent	X Percent of Y, keeping Y on stack	Function (dyadic)	
%MRR	%MRR	Mean rate of return	Mean rate of return in percentage per period (% :)	Command	
%T	%T	Percentage of total	Percentage of total, keeping Y on stack (% :)	Function (dyadic)	
%Σ	%Σ	Percentage of sum	Percentage of x to Σx (% :)	Function (monadic)	
%Σ,Δ%Σ	%Σ,Δ%Σ	Pct of sum and Delta pct to mean	Percentage of x to Σx and Delta percentage to mean using statistics matrix (STATS) (% : ; Δ% : (2 stack levels))	Function (monadic)	
%+MG	%+MG	Add margin to cost	Add margin of X to cost of Y	Function (dyadic)	
√x	√x	Square root	Square root	Function (monadic)	
√(1+x ²)	√(1+x ²)	√(1+x ²)	√(1+x ²)	Function (legacy ; dyadic)	
∫	∫	Integral	Calculates integral (∫z)	Command	
∠	∠	Argument (angle)	Argument (angle) of complex number	Function (monadic)	
▢ADV	▢ADV	Printer advance	Print buffer followed by linefeed (advance)	Command (strike)	
▢ALLr	▢ALLr	Print all registers	Print all registers - save contents as text to data file in FAT (Filename format YYYYMMDD-HHMMSS00.REGS)	Command	
▢CHAR	▢CHAR	Print character	Print single character	Command (strike)	
▢DLAY	▢DLAY	Printer delay	Set printer delay	Command (strike)	
▢LCD	▢LCD	Print LCD	Print screen contents	Command (strike)	
▢MODE	▢MODE	Printer mode	Set printer mode (0: printer font ; 1: variable ; 2: small ; 3: ascii)	Command (strike)	
▢PROG	▢PROG	Print program	Print current program (to text file in FAT) (▢PROG _ TamLabel ; DMCP : File save dialog (PROGRAMS))	Command	
▢R	▢R	Print register	Print register	Command (strike)	
▢REGS	▢REGS	Print registers	Print registers (sss.nn means print registers from sss through sss + nn - 1)	Command (strike)	
▢STK	▢STK	Print stack	Print stack registers - save contents as text to data file in FAT (Filename format YYYYMMDD-HHMMSS00.REGS)	Command	
▢TAB	▢TAB	Print column	Position print head to column	Command (strike)	
▢USER	▢USER	Print user items	Print user items: variable names and programs	Command (strike)	
▢WIDTH	▢WIDTH	Print width	Number of print columns or pixels (depending on ▢MODE)	Command (strike)	
▢X	▢X	Print X	Print stack register X	Command (strike)	
▢Σ	▢Σ	Print sums	Print summation registers	Command (strike)	
▢*	▢*	Print byte	Print single byte	Command (strike)	
#B	#B	Number of bits set	Count number of bits set	Function (monadic)	